AI Architecture, Behaviors & Interaction Design

Project Name: AI Patrol and Chase System in Unity

1. Overview of the AI System

The AI system in this Unity project is designed to simulate intelligent enemy behavior that reacts to the player's presence. The system includes three main behaviors: patrol, chase, and search. It uses Unity's built-in NavMesh system for pathfinding and Raycasting for player detection.

The AI is implemented using a simple state machine that switches between these behaviors based on the player's position and visibility.

2. AI Behaviors Implemented

Patrolling

- The AI follows a set of waypoints when idle.
- This allows the enemy to look like it's "guarding" an area when it hasn't seen the player.
- Waypoints can be set using empty GameObjects and assigned in the script.

Chasing

- When the player enters the AI's detection range and is visible (no obstacles in between), the AI switches to Chase mode.
- The AI follows the player using the NavMeshAgent, constantly updating its destination to the player's current position.

Searching

- If the player moves out of sight (e.g., hides behind a wall), the AI goes to the last known position of the player.
- It then searches for a few seconds, simulating awareness.
- If the player is not found, the AI returns to Patrolling.

3. Technical Components Used

Component	Purpose
NavMesh & NavMeshAgent	Handles pathfinding and smooth movement over the terrain.
Raycasting	Used to simulate the AI's "line of sight" to detect the player.
State Machine (Script)	Manages AI behavior switching (Patrol → Chase → Search → Patrol).
Colliders & Rigidbody	Ensure physical interactions and prevent overlapping.
Obstacles (NavMeshObstacle)	Prevent AI from walking through walls or objects.

4. Interaction Design

- The player uses **WASD/Arrow Keys** to move.
- The enemy reacts based on the player's **visibility** and **distance**.
- Obstacles in the scene allow the player to **hide**, which affects AI behavior.
- All movement is handled through Unity's **physics system** and AI pathfinding.

5. Final Behavior Testing Checklist

Test Scenario	Expected Behavior	Result
AI patrols when idle	Patrols between waypoints	✓
AI chases when player is seen	Follows player accurately using NavMeshAgent	✓
AI searches when player hides	Moves to last known position, searches, returns to patrol	✓
AI avoids obstacles	Navigates around walls/boxes using NavMesh	✓
AI stops at player (no clipping)	AI does not walk through the player	✓

6. Conclusion

This project demonstrates the foundation of a functional AI system in Unity using event-driven logic, navigation, and player interaction. The system can be further extended with animations, attack logic, stealth mechanics, and more complex states.